L5: Digital filters

Linear time invariant systems
Impulse response
Transfer function
Digital filter analysis
Example: speech synthesis

This lecture is based on chapter 10 of [Taylor, TTS synthesis, 2009]
Filters

A filter is a mathematical model of a system used for modifying signals

- In some applications, one is interested in “filtering out” unwanted portions of a signal
- Our interest in filters here comes from the acoustic theory of speech
  - According to the “source-filter” model, speech is a process by which a glottal source is modified by a vocal tract filter
Linear time invariant (LTI) filters

- A class of linear filters whose behavior does not change over time
  - Linearity implies that the filter meets the scaling and superposition properties
    \[ x[n] \rightarrow y[n] \implies \alpha x_1[n] + \beta x_2[n] \rightarrow \alpha y_1[n] + \beta y_2[n] \]
- LTI filters are generally described in terms of difference equations

Types of LTI filters

- Finite impulse response (FIR)
  - Operate only on previous values of the input
    \[ y[n] = \sum_{k=0}^{M} b_k x[n - k] \]
- Infinite impulse response (IIR)
  - Operate as well on previous values of the output
    \[ y[n] = \sum_{k=0}^{M} b_k x[n - k] + \sum_{l=0}^{N} a_l y[n - l] \]
http://www.mikroe.com/eng/chapters/view/73/chapter-3-iir-filters/
The impulse response

- The properties of a filter in the time domain can be described by its response when the input is an impulse

\[ \delta[n] = \begin{cases} 
0 & n = 0 \\
1 & n \neq 0
\end{cases} \]

- Consider the IIR filter defined by \( y[n] = x[n] - 0.8y[n - 1] \)
  - Impulse response has no fixed duration (it is infinite, hence the name)
  - The response is an exponential decay controlled by \( a_1 = -0.8 \)
    - For \( a_1 > 1 \), output grows exponentially, and the filter is said to be unstable

- Now consider the IIR filter \( y[n] = -1.8y[n - 1] + y[n - 2] \)
  - In this case, the response has the shape of a sine wave

- Finally, consider the IIR filter \( y[n] = -1.78y[n - 1] + 0.9y[n - 2] \)
  - In this case, the response has the shape of a decaying sine wave, a mix of the previous two signals

- Thus, the response characteristics are entirely defined by the parameters of the filter
Example

ex5p1.m

- Generate example of IIR and FIR filters
- Show how the impulse response is infinite for IIR but finite for FIR
  (examples from Taylor §10.4.1-2)
The filter convolution sum

- If we know the impulse response $h[n]$ of a filter, its response to any input sequence $x[n]$ can be computed as

$$y[n] = \sum_{k} x[k] h[n - k]$$

The filter transfer function

- The impulse response describes the filter properties in the time domain
- We will now see how to describe the filter in the frequency domain
- Consider the generic IIR filter

$$y[n] = b_0 x[n] + b_1 x[n - 1] + \cdots + b_M x[n - M] + a_1 y[n - 1] + a_2 y[n - 2] + a_N y[n - N]$$

- And let’s apply the Z transform

$$Y(z) = b_0 X(z) + b_1 X(z)z^{-1} + \cdots + b_M X(z)z^{-M} + a_1 Y(z)z^{-1} + \cdots + a_N Y(z)z^{-N}$$
which, grouping terms, can be expressed as

\[ Y(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{M-1}}{1 - a_1 z^{-1} - \cdots - a_N z^{N-1}} X(z) \]

from which the transfer function of the filter can be defined as:

\[ H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{l=0}^{N} a_l z^{-l}} \]

\[ \text{NOTES} \]

– As we will see in the next few slides, the transfer function \( H(z) \) fully defines the filter’s characteristics in the frequency domain

– It can be shown that the transfer function is the Z-transform of the impulse response \( H(z) = \sum h[k] z^{-k} \)

– The transfer function is a ratio of two polynomials whose coefficients are those of the difference equation
Filter analysis and design

Filter analysis

– The coefficients of first order filters are readily interpretable, for example as the rates of decay of exponentials
– For higher-order filters, interpretation of the coefficients is very hard
– Instead, we employ polynomial analysis to produce an easier interpretation of the transfer function

Polynomial analysis and design

– Consider the quadratic expression \( f(x) = 2x^2 - 6x + 1 \)
  • This equation can be factorized as \( f(x) = G(x - q_1)(x - q_2) \), where \( (q_1, q_2) \) are the roots of the expression and \( G \) is the gain
  • The roots \( (q_1, q_2) \) are called the zeros because \( f(q_i) = 0 \)

– Now consider the inverse filter function \( f(x) = \frac{1}{2x^2 - 6x + 1} \)
  • This curve is very different, and the function “blows up” at \( x = \{q_1, q_2\} \)
  • The roots \( (q_1, q_2) \) are called the poles ... maybe because they create a pole-like effect on the curve?
(b) plot of \( g \times (2x^2 - 6x + 1) \) for different values of \( g \)

(a) plot of \( \frac{1}{2x^2 - 6x + 1} \)

[Taylor, 2009]
We can now use polynomials to analyze our filter’s transfer function. Consider the transfer function

\[ H(z) = \frac{1}{z^2 - a_1 z - a_2} \]

Since transfer functions are generally expressed in terms of \( z^{-1} \), we multiply numerator and denominator by \( z^{-2} \) to obtain

\[ H(z) = \frac{z^{-2}}{1 - a_1 z^{-1} - a_2 z^{-2}} = G \frac{z^{-2}}{(1 - p_1 z^{-1})(1 - p_2 z^{-2})} \]

The figures in the next slide show the shape of the transfer function for \( a_1 = 1, a_2 = -0.5 \):

- In this case the roots of the denominator are complex \( 0.5 \pm j0.5 \)
- Note how the shape of the filter can be described by the position of the poles in the Z plane; we do not need to plot \( |H(z)| \)
[Taylor, 2009]
– The same analysis can be extended to any LTI filter

\[ H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^M}{1 - a_1 z^{-1} - \cdots - a_N z^N} \]

– By expressing it in terms of its factors

\[ H(z) = \frac{(1 - q_1 z^{-1})(1 - q_2 z^{-1}) \cdots (1 - q_M z^{-1})}{(1 - p_1 z^{-1})(1 - p_2 z^{-1}) \cdots (1 - p_N z^{-1})} \]

– And then analyzing the position of its poles and zeros in the Z plane
Frequency interpretation of $H(z)$

- Recall that the $z$ transform for the digital signal $x[n]$ is
  
  \[ X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n} \]

- And that its Fourier transform is obtained by making $z = e^{j\hat{\omega}}$
  
  \[ X(e^{j\hat{\omega}}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\hat{\omega}n} \]

- Therefore, you can find the frequency response by substituting $\hat{\omega}$ with the frequency of interest

  - Since $e^{j\hat{\omega}}$ is unit length, this can be thought of as sweeping out a circle of radius 1 in the $z$-domain
  
  - This is consistent with the fact that the spectrum $X(e^{j\hat{\omega}})$ is periodic with period $\hat{\omega} = 2\pi$
[Taylor, 2009]
Filter characteristics

Consider the following first-order IIR filter

\[ h[n] = b_0 x[n] - a_1 y[n - 1] \]

\[ H(z) = \frac{b_0}{1 - a_1 z^{-1}} = \frac{b_0}{1 - a_1 e^{-j\omega}} \]

The figures in the next page show the time- and frequency domain response, pole locations and pole locations in the z-domain for \( b_1 = 1 \) and \( a_1 = \{0.8, 0.7, 0.6, 0.4\} \)

• This type of filter is known as a resonator, and the peak is known as a resonance because frequencies near that peak are amplified by the filter

Analysis

• As the length of the decay increases, the peak becomes sharper
• Large \( a_1 \) corresponds to slow decays and narrow bandwidths
• Small \( a_1 \) corresponds to fast decays and broad bandwidths
[Taylor, 2009]
– Resonances are generally described by three properties: amplitude, frequency, and bandwidth
  • The radius of the pole controls the amplitude and bandwidth
  • The angle of the pole controls the frequency; in this case $\hat{\omega} = 0$ since the pole lies on the real line
– In order to model speech resonances at non-zero frequencies, we then move the pole away from the real axis
  • This will result in a complex pole $p_1 = re^{j\theta} = \alpha + j\beta$, which leads to a complex filter coefficient $a_1$; see next slide
  • For this reason, we introduce complex-conjugate pairs of poles $re^{\pm j\theta}$

$$H(z) = \frac{1}{(1 - re^{j\theta}z^{-1})(1 - re^{-j\theta}z^{-1})} = \frac{1}{1 - 2rcos(\theta)z^{-1} + r^2z^{-1}}$$

– Examples for various pole positions are shown in the next slide
  • For constant $\theta$, the filter becomes sharper as $r \to 1$
    – For small $r$, the skirts of the two poles overlap and shift the resonance
  • For constant $r$, resonances move away from $\hat{\omega} = 0$ as $\theta \to 1$
Taylor, 2009
Effect of zeros

- Adding a term $b_1 = 1$ places a zero at the origin
- Adding a term $b_1 = -1$ places a zero at the ends of the spectrum

Thus, zeros add anti-resonances to the spectrum
– Thus, we can build any transfer function by placing poles and zeros at the appropriate locations and then multiplying their transfer functions

– Note, though, that poles that are close together will interact, so the final resonances of a system cannot always be predicted from their poles
Example

Let’s now use an LTI filter to synthesize English vowel [ih]

– Remember that normal frequency $F$ (Hz) can be converted into normalized frequency $\hat{\omega} = \frac{2\pi F}{F_S}$

– From this expression we can calculate pole positions as
  \[
  \theta = \frac{2\pi F}{F_S} \\
  r = e^{-\pi B/F_S}
  \]

– From acoustic phonetics, we can estimate formant values for [ih] to be
  \[
  \{F_1, F_2, F_3\} = \{300Hz, 2200Hz, 3000Hz\}
  \]

– Formant bandwidths are harder to measure, so we assume all three to be equal to $B = 250Hz$

– Assuming a sampling frequency of $F_S = 16kHz$, this results in

<table>
<thead>
<tr>
<th>Formant</th>
<th>Frequency (Hz)</th>
<th>Bandwidth (Hz)</th>
<th>$r$</th>
<th>$\theta$ (normalised angular frequency)</th>
<th>pole</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>300</td>
<td>250</td>
<td>0.95</td>
<td>0.12</td>
<td>0.963 + 0.116j</td>
</tr>
<tr>
<td>F2</td>
<td>2200</td>
<td>250</td>
<td>0.95</td>
<td>0.86</td>
<td>0.619 + 0.719j</td>
</tr>
<tr>
<td>F3</td>
<td>3000</td>
<td>250</td>
<td>0.95</td>
<td>1.17</td>
<td>0.370 + 0.874j</td>
</tr>
</tbody>
</table>
The transfer function for each formant can be estimated as

$$H_n(z) = \frac{1}{(1 - p_n z^{-1})(1 - p_n^* z^{-1})}$$

And the complete vocal tract TF can be estimated by multiplication

$$H(z) = H_1(z)H_2(z)H_3(z)$$
Example

```matlab
ex5p2.m
Synthesize speech sample using the previous vocal tract filter and a pulse train as glottal excitation
```